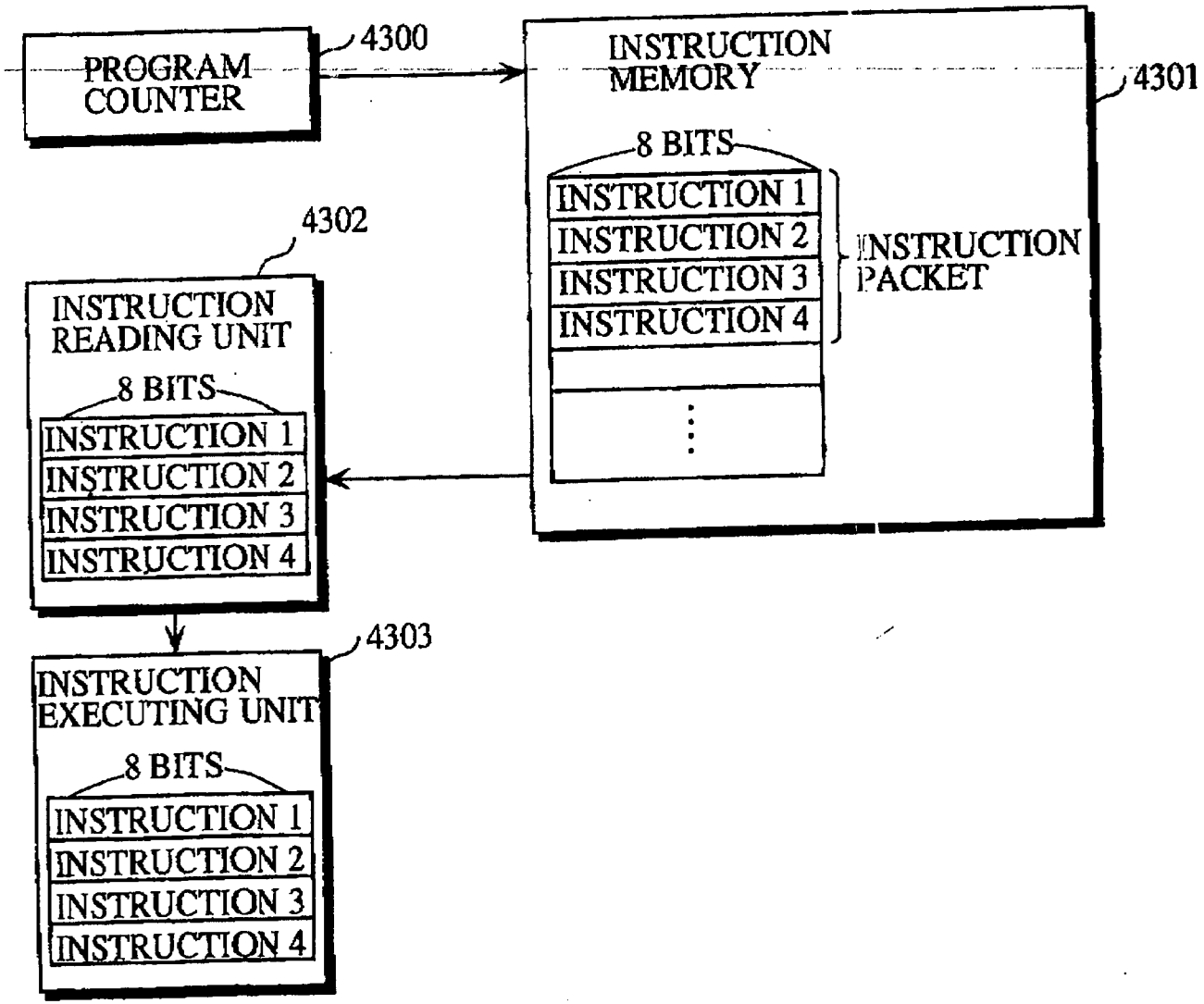
FIG. 1


PROGRAM COUNTER
4300
INSTRUCTION MEMORY
4301
8 BITS
INSTRUCTION 1
INSTRUCTION 2
INSTRUCTION 3
INSTRUCTION 4
INSTRUCTION PACKET
4302
INSTRUCTION READING UNIT
8 BITS
INSTRUCTION 1
INSTRUCTION 2
INSTRUCTION 3
INSTRUCTION 4
4303
INSTRUCTION EXECUTING UNIT
8 BITS
INSTRUCTION 1
INSTRUCTION 2
INSTRUCTION 3
INSTRUCTION 4

PARALLEL EXECUTION BOUNDARY INFORMATION 100

FORMAT INFORMATION 101

FIG. 2A

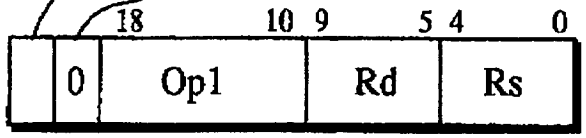


FIG. 2B

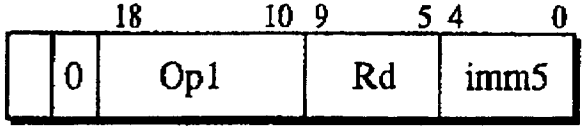


FIG. 2C

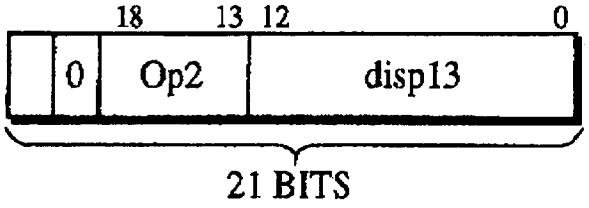


FIG. 2D

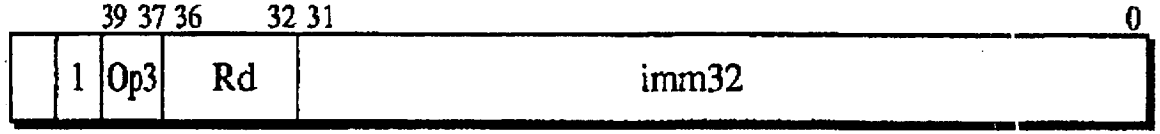


FIG. 2E

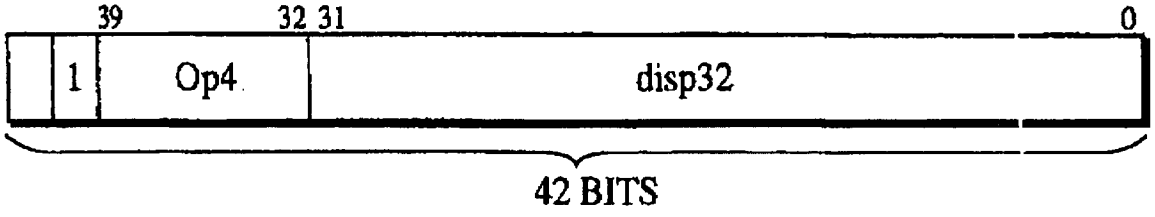


FIG. 3A

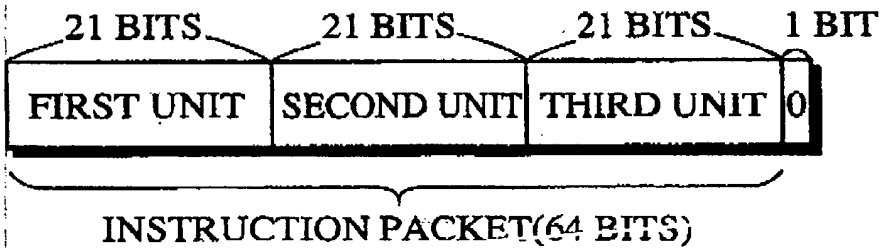


FIG. 3B

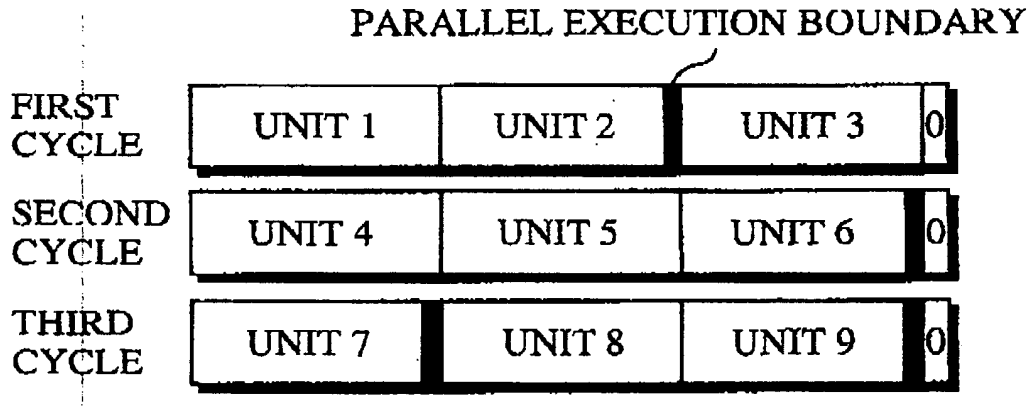


FIG. 3C

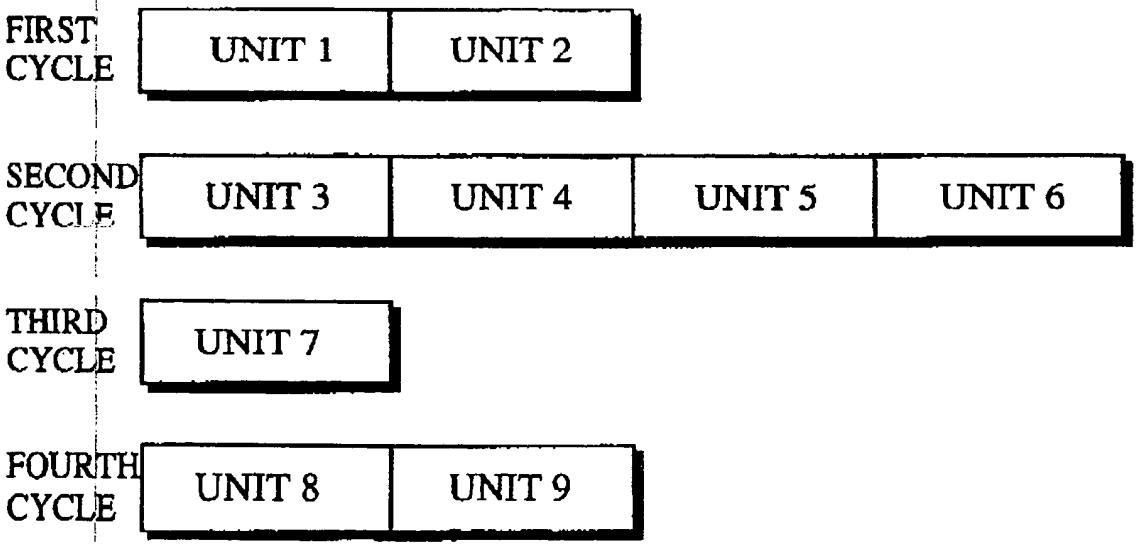


FIG. 4

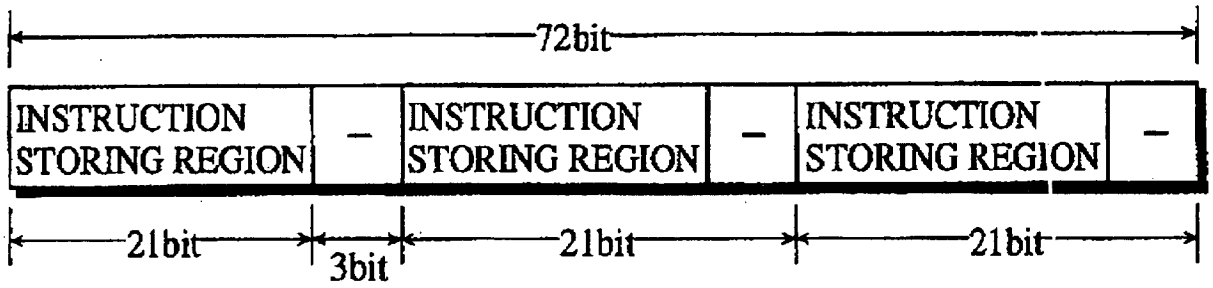

72bit
INSTRUCTION STORING REGION
–
INSTRUCTION STORING REGION
–
INSTRUCTION STORING REGION
–
21bit
3bit
21bit
21bit

FIG. 5

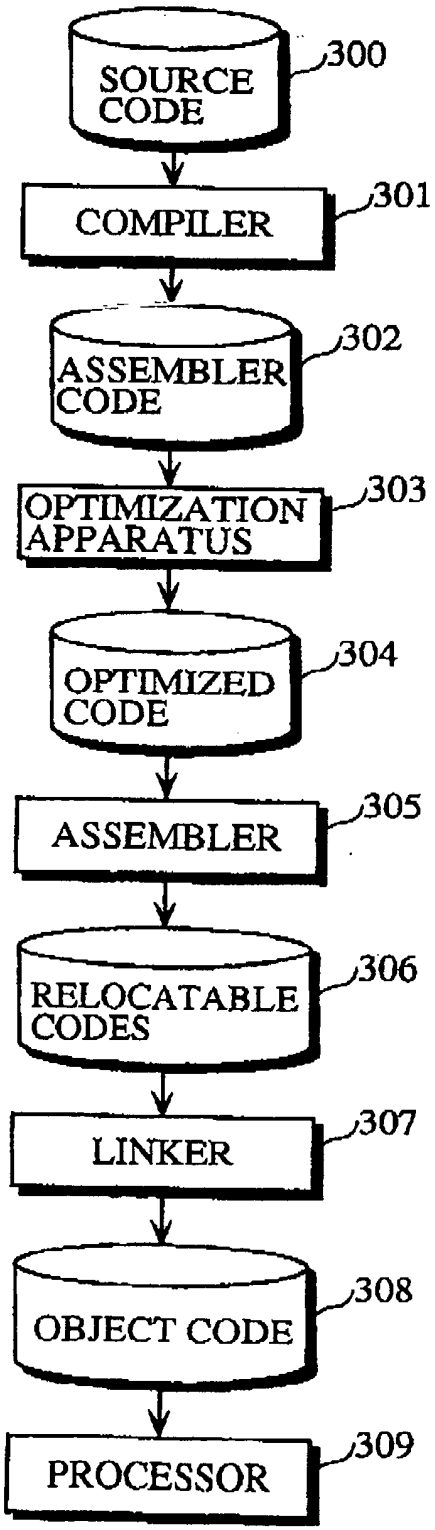

SOURCE CODE
300
COMPILER
301
ASSEMBLER CODE
302
OPTIMIZATION APPARATUS
303
OPTIMIZED CODE
304
ASSEMBLER
305
RELOCATABLE CODES
306
LINKER
307
OBJECT CODE
308
PROCESSOR
309

FIG. 6

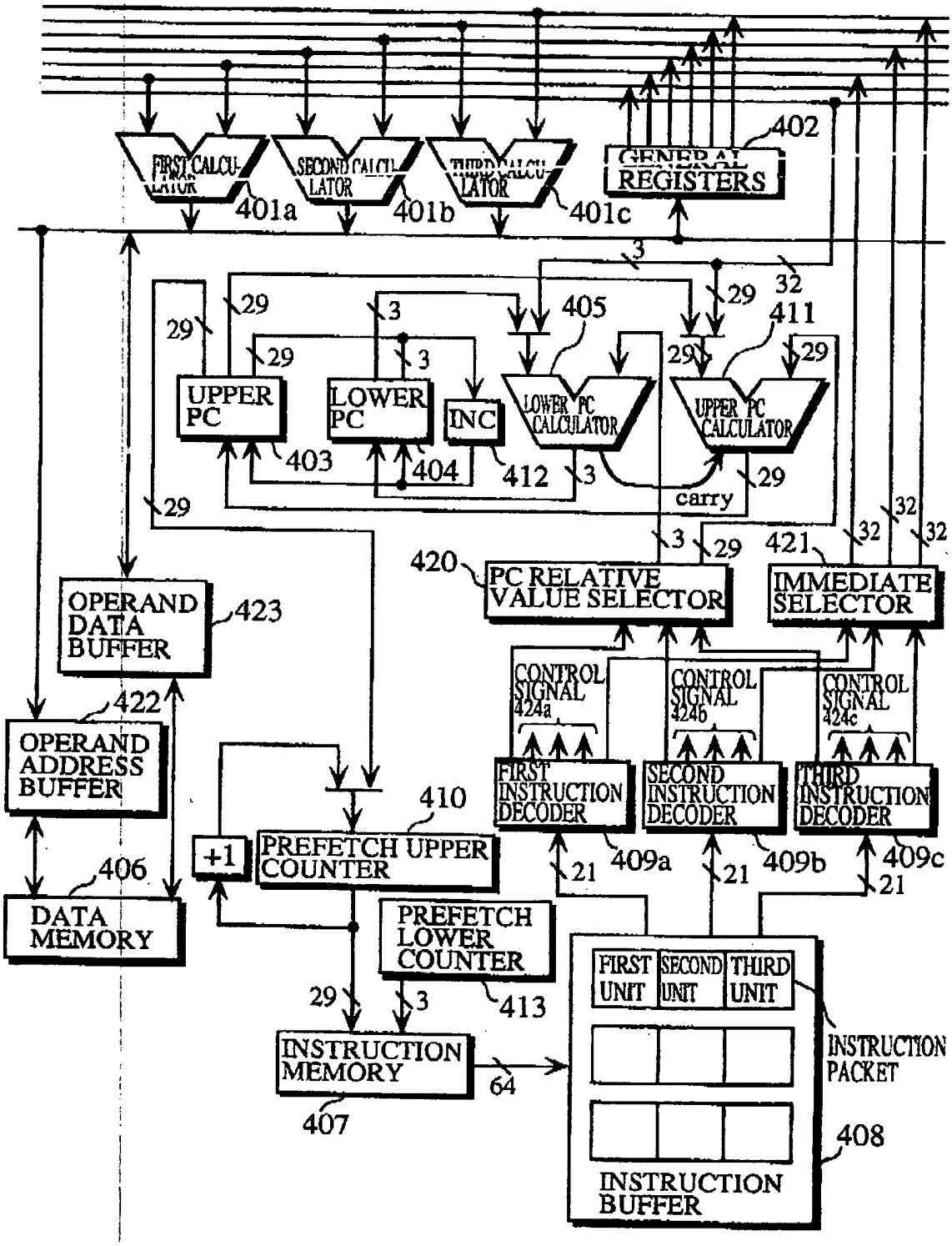

FIRST CALCULATOR
SECOND CALCULATOR
THIRD CALCULATOR
401a
401b
401c
GENERAL REGISTERS
402
UPPER PC
LOWER PC
INC
403
404
412
LOWER PC CALCULATOR
UPPER PC CALCULATOR
405
411
carry
OPERAND DATA BUFFER
423
OPERAND ADDRESS BUFFER
422
DATA MEMORY
406
PC RELATIVE VALUE SELECTOR
420
IMMEDIATE SELECTOR
421
CONTROL SIGNAL 424a
CONTROL SIGNAL 424b
CONTROL SIGNAL 424c
FIRST INSTRUCTION DECODER
SECOND INSTRUCTION DECODER
THIRD INSTRUCTION DECODER
409a
409b
409c
+1
PREFETCH UPPER COUNTER
410
PREFETCH LOWER COUNTER
413
INSTRUCTION MEMORY
407
64
FIRST UNIT
SECOND UNIT
THIRD UNIT
INSTRUCTION PACKET
INSTRUCTION BUFFER
408

FIG. 7

IN-PACKET ADDRESS BEFORE UPDATING / INCREMENT VALUE	3'b000	3'b010	3'b100
1	3'b010	3'b100	3'b000 (CARRY 1)
2	3'b100	3'b000 (CARRY 1)	3'b010 (CARRY 1)
3	3'b000 (CARRY 1)	3'b010 (CARRY 1)	3'b100 (CARRY 1)
4	3'b010 (CARRY 1)	3'b100 (CARRY 1)	3'b000 (CARRY 2)

FIG. 8A

LOWER 3 BITS OF PC RELATIVE VALUE \ LOWER 3 BITS OF ADDRESS VALUE	3'b000	3'b010	3'b100
3'b000	3'b000	3'b010	3'b100
3'b010	3'b010	3'b100	3'b000 (CARRY 1)
3'b100	3'b100	3'b000 (CARRY 1)	3'b010 (CARRY 1)

FIG. 8B

LOWER 3 BITS OF ADDRESS VALUE(BEFORE SUBTRACTION) \ LOWER 3 BITS OF ADDRESS VALUE(TO BE SUBTRACTED)	3'b00	0b010	0b100
3'b000	3'b000	3'b100 (CARRY 1)	3'b010 (CARRY 1)
3'b010	3'b010	3'b000	3'b100 (CARRY 1)
3'b100	3'b100	3'b010	3'b000

FIG. 9

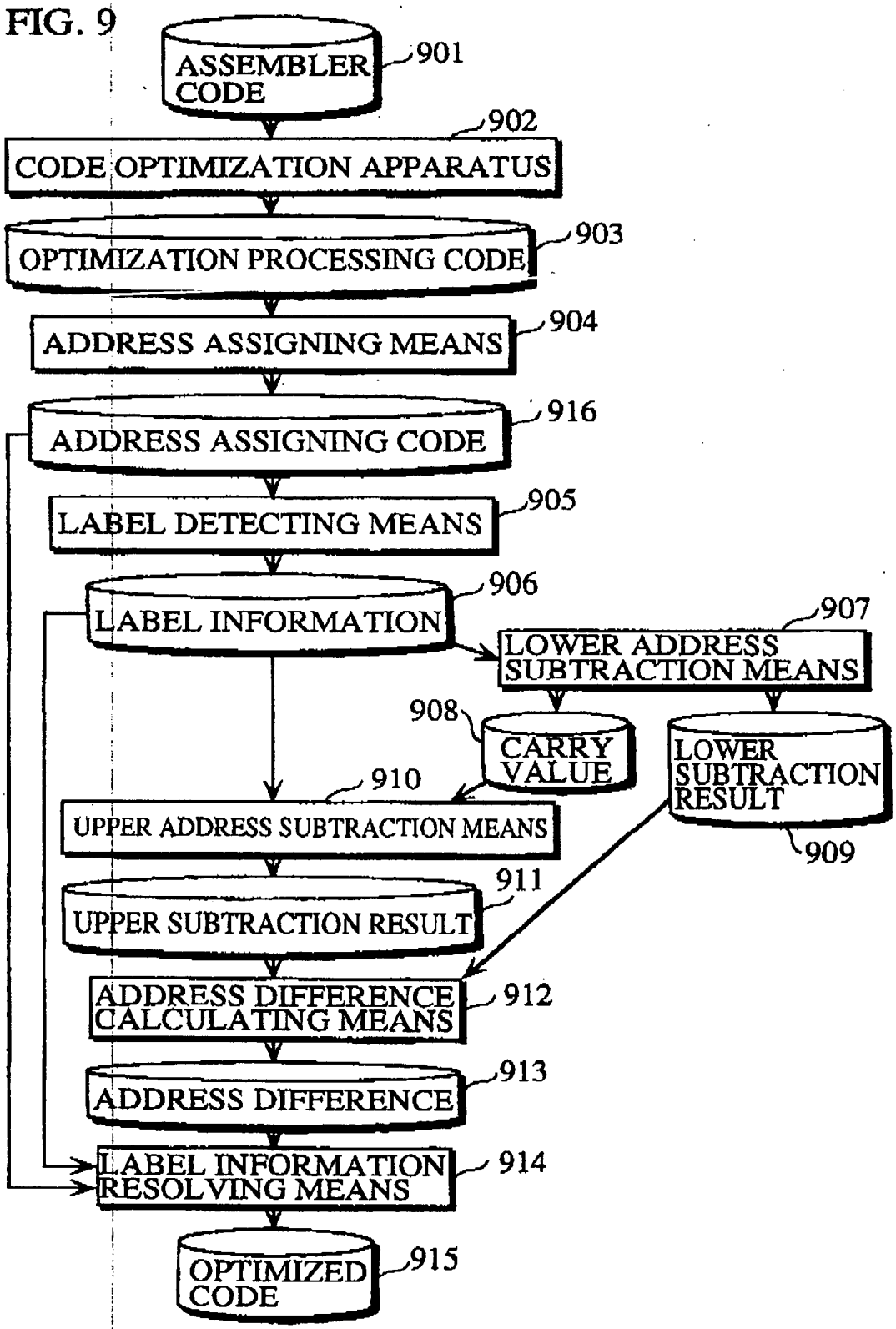

ASSEMBLER CODE
901
CODE OPTIMIZATION APPARATUS
902
OPTIMIZATION PROCESSING CODE
903
ADDRESS ASSIGNING MEANS
904
ADDRESS ASSIGNING CODE
916
LABEL DETECTING MEANS
905
LABEL INFORMATION
906
LOWER ADDRESS SUBTRACTION MEANS
907
CARRY VALUE
908
LOWER SUBTRACTION RESULT
909
UPPER ADDRESS SUBTRACTION MEANS
910
UPPER SUBTRACTION RESULT
911
ADDRESS DIFFERENCE CALCULATING MEANS
912
ADDRESS DIFFERENCE
913
LABEL INFORMATION RESOLVING MEANS
914
OPTIMIZED CODE
915

FIG. 10

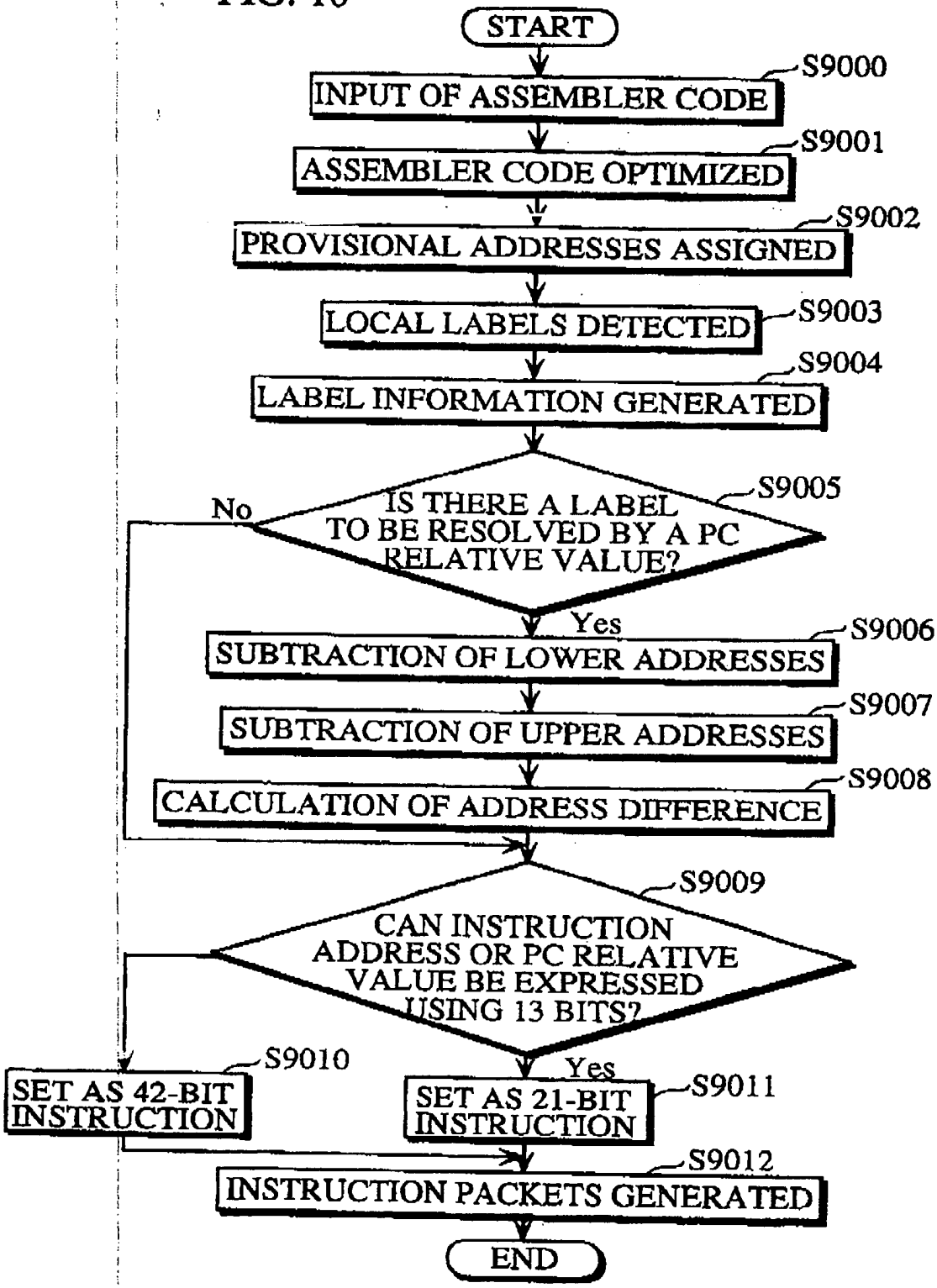

START
INPUT OF ASSEMBLER CODE
S9000
ASSEMBLER CODE OPTIMIZED
S9001
PROVISIONAL ADDRESSES ASSIGNED
S9002
LOCAL LABELS DETECTED
S9003
LABEL INFORMATION GENERATED
S9004
IS THERE A LABEL TO BE RESOLVED BY A PC RELATIVE VALUE?
S9005
No
Yes
SUBTRACTION OF LOWER ADDRESSES
S9006
SUBTRACTION OF UPPER ADDRESSES
S9007
CALCULATION OF ADDRESS DIFFERENCE
S9008
CAN INSTRUCTION ADDRESS OR PC RELATIVE VALUE BE EXPRESSED USING 13 BITS?
S9009
Yes
SET AS 42-BIT INSTRUCTION
S9010
SET AS 21-BIT INSTRUCTION
S9011
INSTRUCTION PACKETS GENERATED
S9012
END

FIG. 11

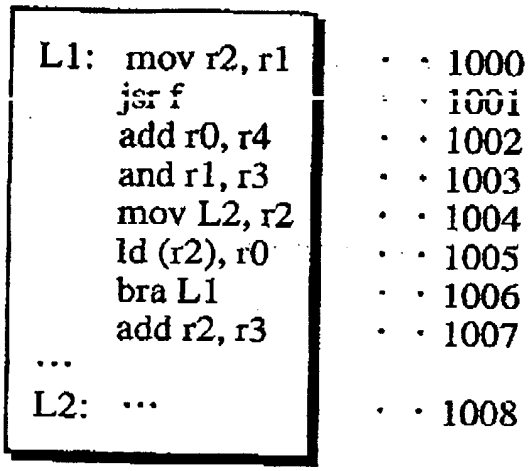


FIG. 12

32'h00000800	L1: mov r2, r1	1000
32'h00000802	jsr f	1001
32'h00000804	add r0, r4	1002
32'h00000808	and r1, r3	1003
32'h0000080a	mov L2, r2	1004
32'h00000810	ld (r2), r0	1005
32'h00000812	bra L1	1006
32'h00000814	add r2, r3	1007
	...	
32'h12345678	L2: ...	1008

FIG. 13

INSTRUCTION	RESOLVING VALUE	
mov L2, r2	ADDRESS	32'h12345678
bra L1	PC RELATIVE VALUE	32'h00000800-32'h00000812

FIG. 14

```
L1:  mov r2, r1  ||  jsr f          ||  add r0, r4      ··1300
     and r1, r3  ||  mov L2, r2     ||  (mov L2, r2)    ··1301
     ld (r2) , r0 ||  bra L1        ||  add r2, r3      ··1302
     ...
L2:                                                     ··1303
```

FIG. 15

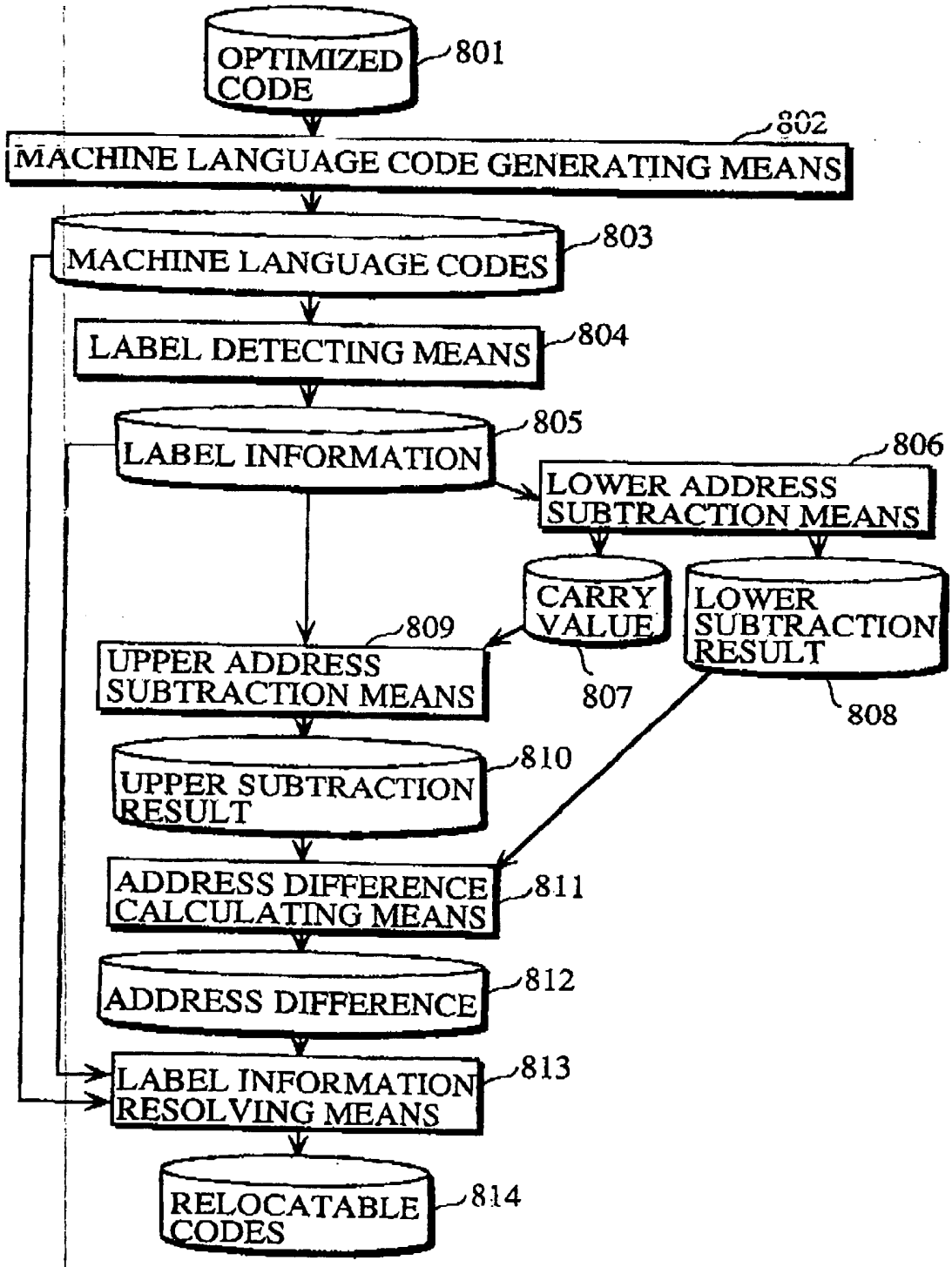

OPTIMIZED CODE
801
MACHINE LANGUAGE CODE GENERATING MEANS
802
MACHINE LANGUAGE CODES
803
LABEL DETECTING MEANS
804
LABEL INFORMATION
805
LOWER ADDRESS SUBTRACTION MEANS
806
CARRY VALUE
807
LOWER SUBTRACTION RESULT
808
UPPER ADDRESS SUBTRACTION MEANS
809
UPPER SUBTRACTION RESULT
810
ADDRESS DIFFERENCE CALCULATING MEANS
811
ADDRESS DIFFERENCE
812
LABEL INFORMATION RESOLVING MEANS
813
RELOCATABLE CODES
814

FIG. 16

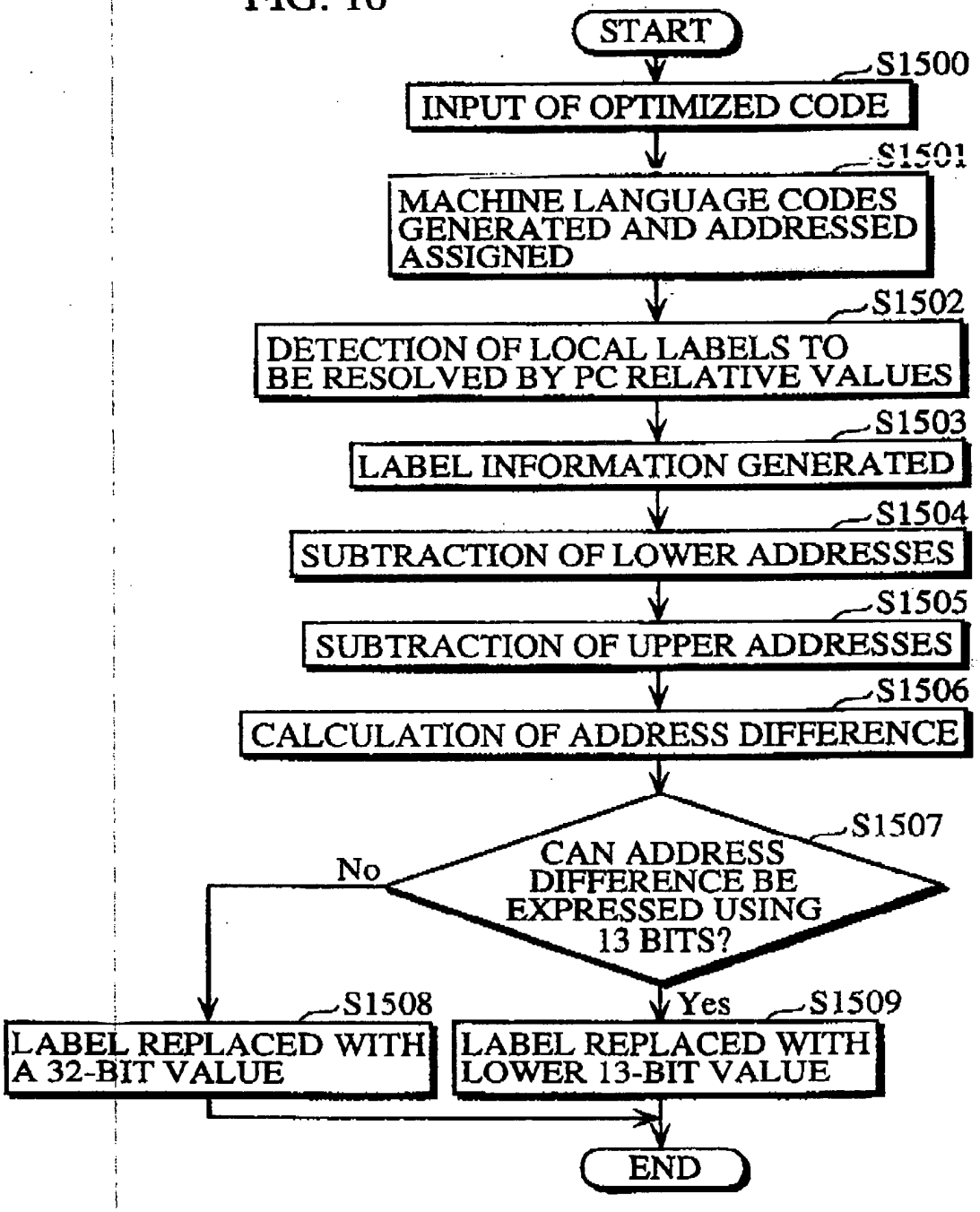

START
S1500
INPUT OF OPTIMIZED CODE
S1501
MACHINE LANGUAGE CODES GENERATED AND ADDRESSED ASSIGNED
S1502
DETECTION OF LOCAL LABELS TO BE RESOLVED BY PC RELATIVE VALUES
S1503
LABEL INFORMATION GENERATED
S1504
SUBTRACTION OF LOWER ADDRESSES
S1505
SUBTRACTION OF UPPER ADDRESSES
S1506
CALCULATION OF ADDRESS DIFFERENCE
S1507
CAN ADDRESS DIFFERENCE BE EXPRESSED USING 13 BITS?
No
Yes
S1508
LABEL REPLACED WITH A 32-BIT VALUE
S1509
LABEL REPLACED WITH LOWER 13-BIT VALUE
END

FIG. 17

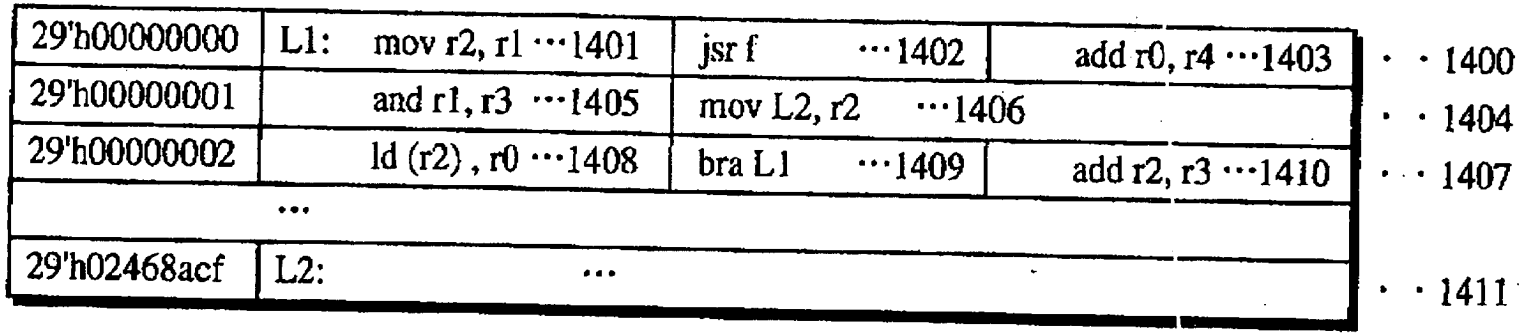

29'h00000000
L1: mov r2, r1 ···1401
jsr f ···1402
add r0, r4 ···1403
· · 1400
29'h00000001
and r1, r3 ···1405
mov L2, r2 ···1406
· · 1404
29'h00000002
ld (r2) , r0 ···1408
bra L1 ···1409
add r2, r3 ···1410
· · · 1407
...
29'h02468acf
L2: ...
· · 1411

FIG. 18

INSTRUCTION	RESOLVING VALUE
bra L1	PC RELATIVE VALUE 32'h00000000-32'h00000012

FIG. 19

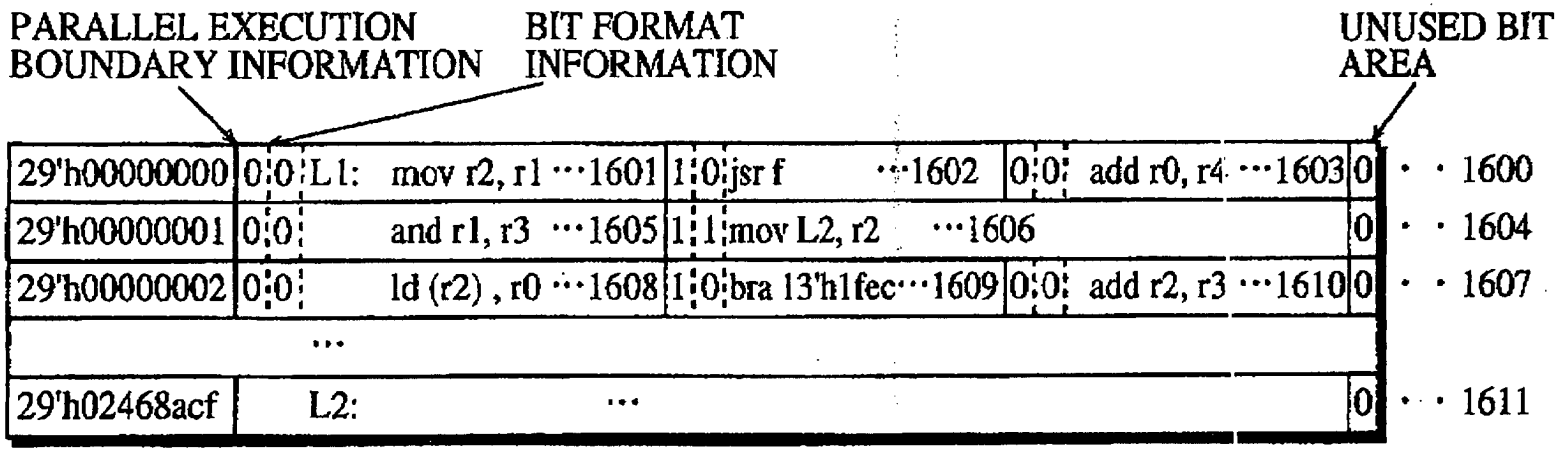

PARALLEL EXECUTION BOUNDARY INFORMATION
BIT FORMAT INFORMATION
UNUSED BIT AREA
29'h00000000 0 0 L1: mov r2, r1 …1601 1 0 jsr f …1602 0 0 add r0, r4 …1603 0 · · 1600
29'h00000001 0 0 and r1, r3 …1605 1 1 mov L2, r2 …1606 0 · · 1604
29'h00000002 0 0 ld (r2) , r0 …1608 1 0 bra 13'h1fec …1609 0 0 add r2, r3 …1610 0 · · 1607
…
29'h02468acf L2: … 0 · · 1611

FIG. 20

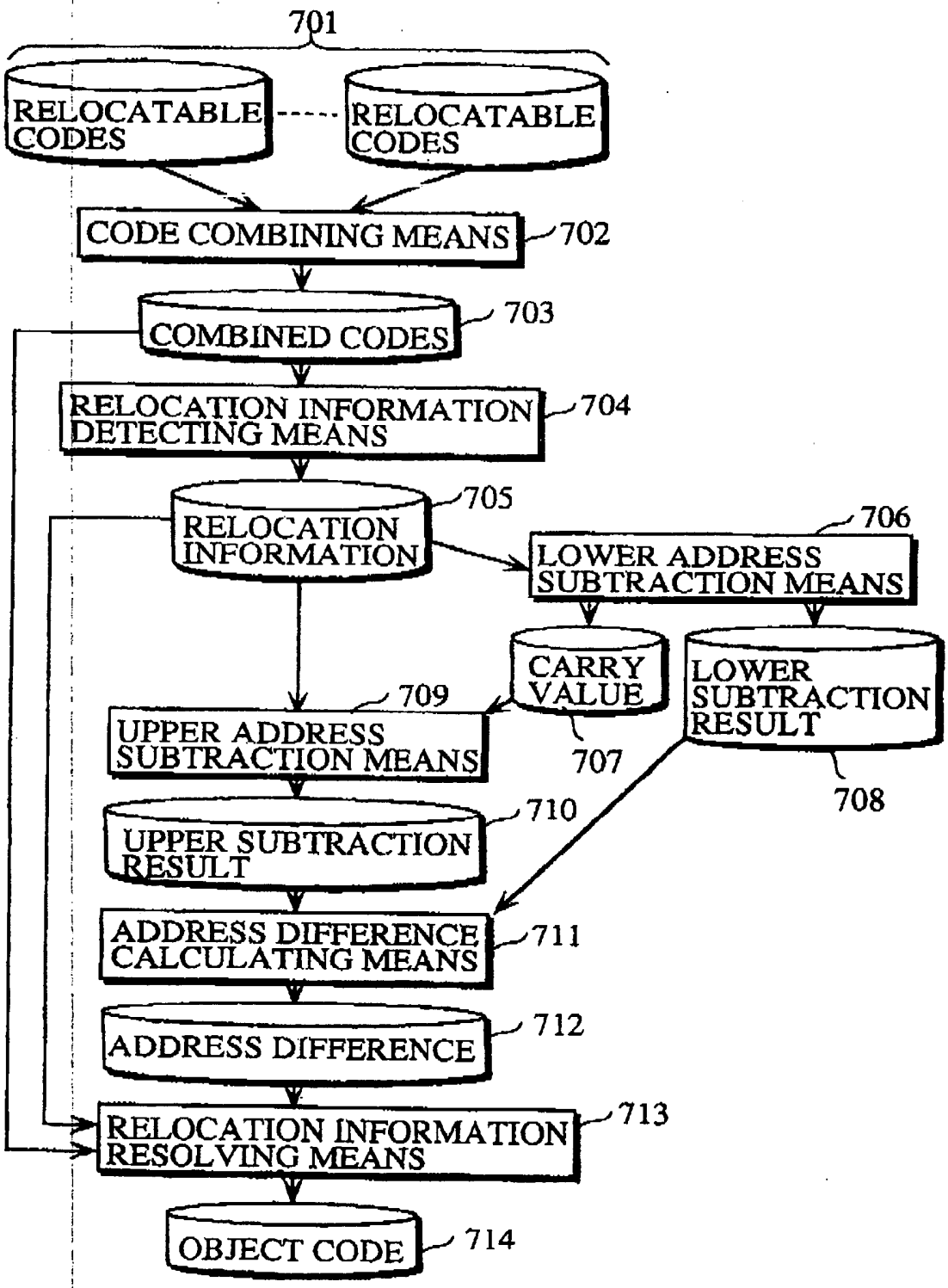

701
RELOCATABLE CODES
RELOCATABLE CODES
CODE COMBINING MEANS
702
COMBINED CODES
703
RELOCATION INFORMATION DETECTING MEANS
704
RELOCATION INFORMATION
705
LOWER ADDRESS SUBTRACTION MEANS
706
CARRY VALUE
707
LOWER SUBTRACTION RESULT
708
UPPER ADDRESS SUBTRACTION MEANS
709
UPPER SUBTRACTION RESULT
710
ADDRESS DIFFERENCE CALCULATING MEANS
711
ADDRESS DIFFERENCE
712
RELOCATION INFORMATION RESOLVING MEANS
713
OBJECT CODE
714

FIG. 21

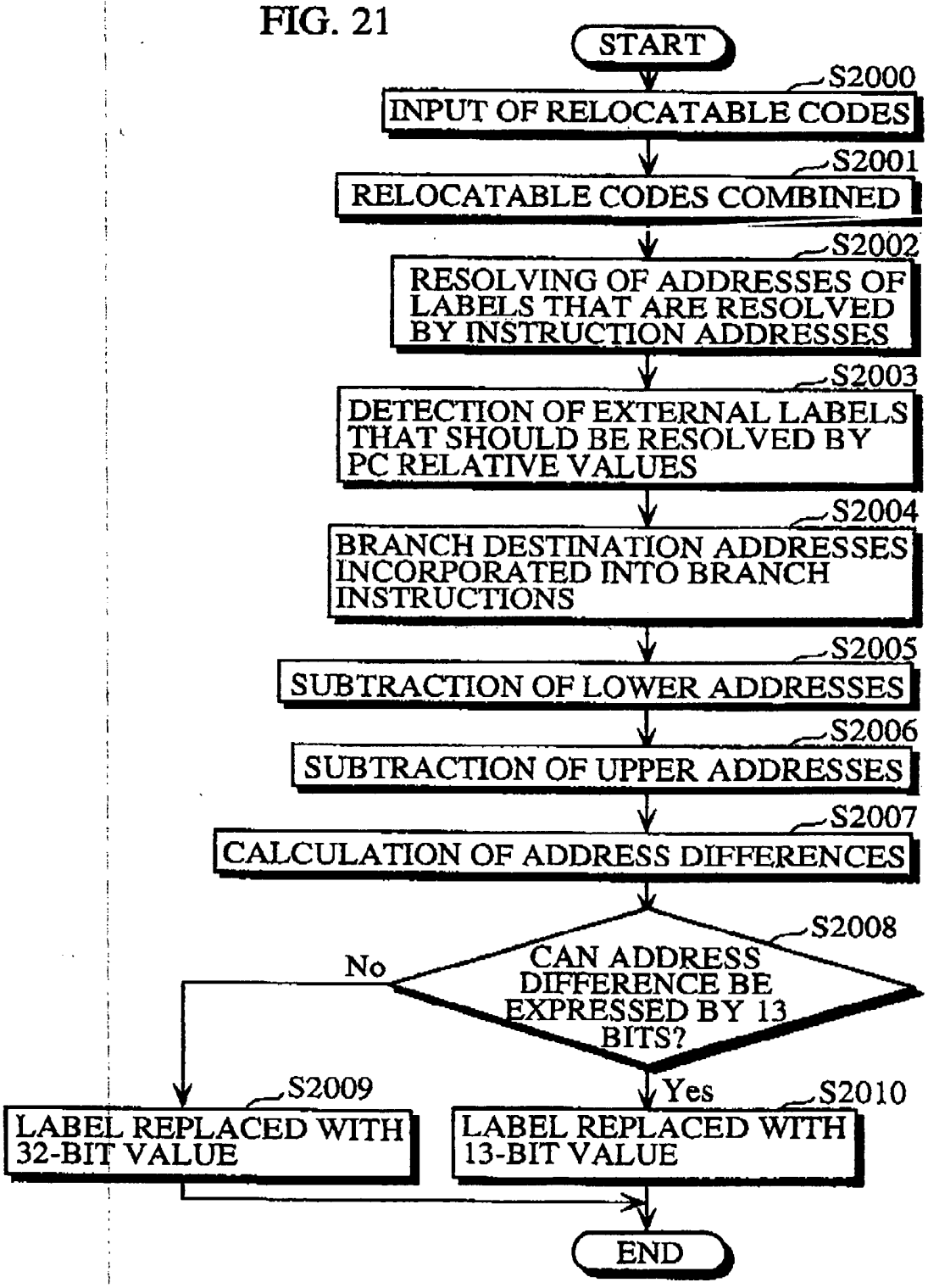

START
S2000
INPUT OF RELOCATABLE CODES
S2001
RELOCATABLE CODES COMBINED
S2002
RESOLVING OF ADDRESSES OF LABELS THAT ARE RESOLVED BY INSTRUCTION ADDRESSES
S2003
DETECTION OF EXTERNAL LABELS THAT SHOULD BE RESOLVED BY PC RELATIVE VALUES
S2004
BRANCH DESTINATION ADDRESSES INCORPORATED INTO BRANCH INSTRUCTIONS
S2005
SUBTRACTION OF LOWER ADDRESSES
S2006
SUBTRACTION OF UPPER ADDRESSES
S2007
CALCULATION OF ADDRESS DIFFERENCES
S2008
CAN ADDRESS DIFFERENCE BE EXPRESSED BY 13 BITS?
No
Yes
S2009
LABEL REPLACED WITH 32-BIT VALUE
S2010
LABEL REPLACED WITH 13-BIT VALUE
END

FIG. 22

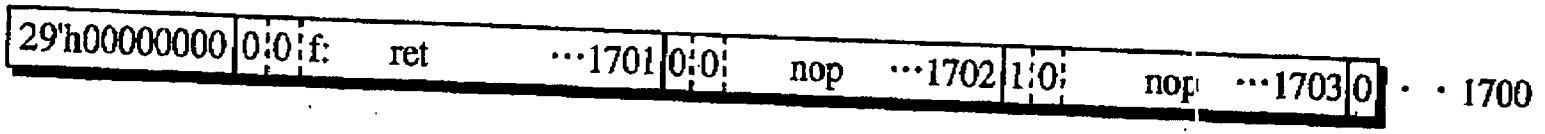


FIG. 23

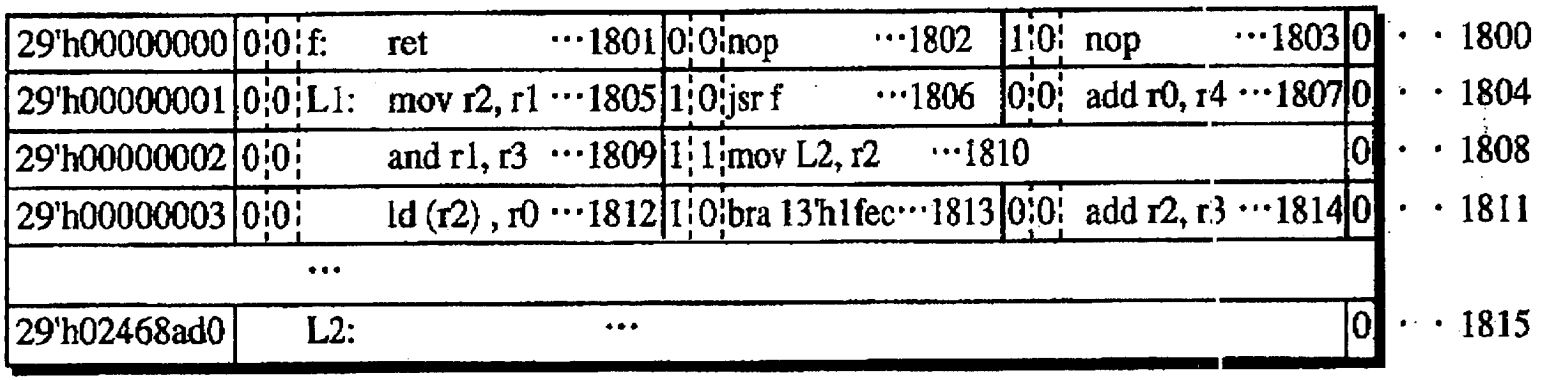

29'h00000000 0 0 f: ret …1801 0 0 nop …1802 1 0 nop …1803 0 · · · 1800
29'h00000001 0 0 L1: mov r2, r1 …1805 1 0 jsr f …1806 0 0 add r0, r4 …1807 0 · · · 1804
29'h00000002 0 0 and r1, r3 …1809 1 1 mov L2, r2 …1810 0 · · · 1808
29'h00000003 0 0 ld (r2) , r0 …1812 1 0 bra 13'h1fec …1813 0 0 add r2, r3 …1814 0 · · · 1811
…
29'h02468ad0 L2: … 0 · · · 1815

FIG. 24

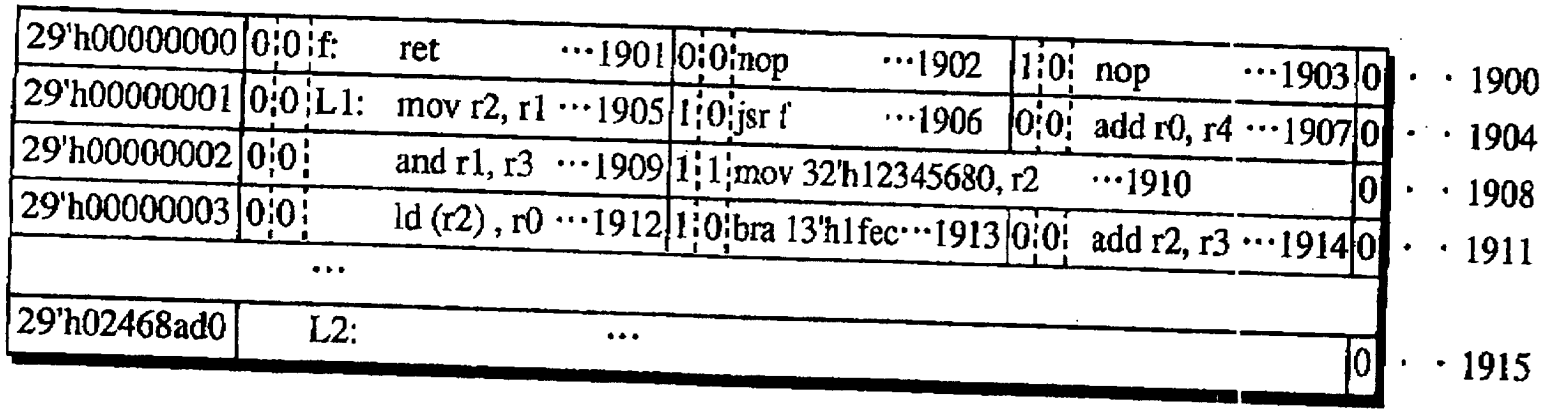

29'h00000000 0 0 f: ret ···1901 0 0 nop ···1902 1 0 nop ···1903 0 · · · 1900
29'h00000001 0 0 L1: mov r2, r1 ···1905 1 0 jsr f ···1906 0 0 add r0, r4 ···1907 0 · · · 1904
29'h00000002 0 0 and r1, r3 ···1909 1 1 mov 32'h12345680, r2 ···1910 0 · · · 1908
29'h00000003 0 0 ld (r2) , r0 ···1912 1 0 bra 13'h1fec···1913 0 0 add r2, r3 ···1914 0 · · · 1911
···
29'h02468ad0 L2: ··· 0 · · · 1915

FIG. 25

INSTRUCTION	RESOLVING VALUE
jsr f	PC RELATIVE VALUE 32'h00000000-32'h0000000a

FIG. 26

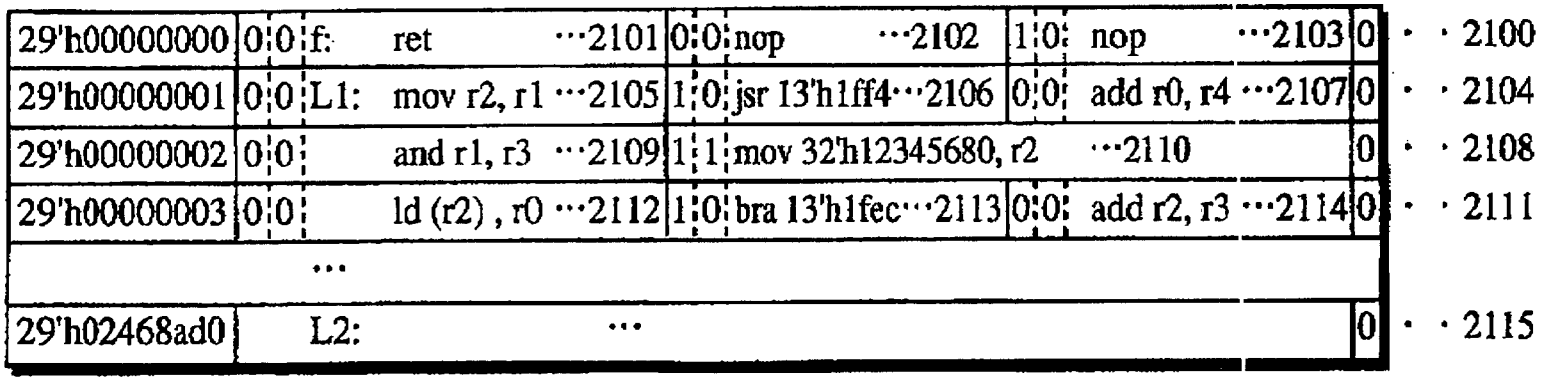

29'h00000000 0 0 f: ret ···2101 0 0 nop ···2102 1 0 nop ···2103 0 · · · 2100
29'h00000001 0 0 L1: mov r2, r1 ···2105 1 0 jsr 13'h1ff4 ···2106 0 0 add r0, r4 ···2107 0 · · · 2104
29'h00000002 0 0 and r1, r3 ···2109 1 1 mov 32'h12345680, r2 ···2110 0 · · · 2108
29'h00000003 0 0 ld (r2) , r0 ···2112 1 0 bra 13'h1fec ···2113 0 0 add r2, r3 ···2114 0 · · · 2111
···
29'h02468ad0 L2: ··· 0 · · · 2115

FIG. 27

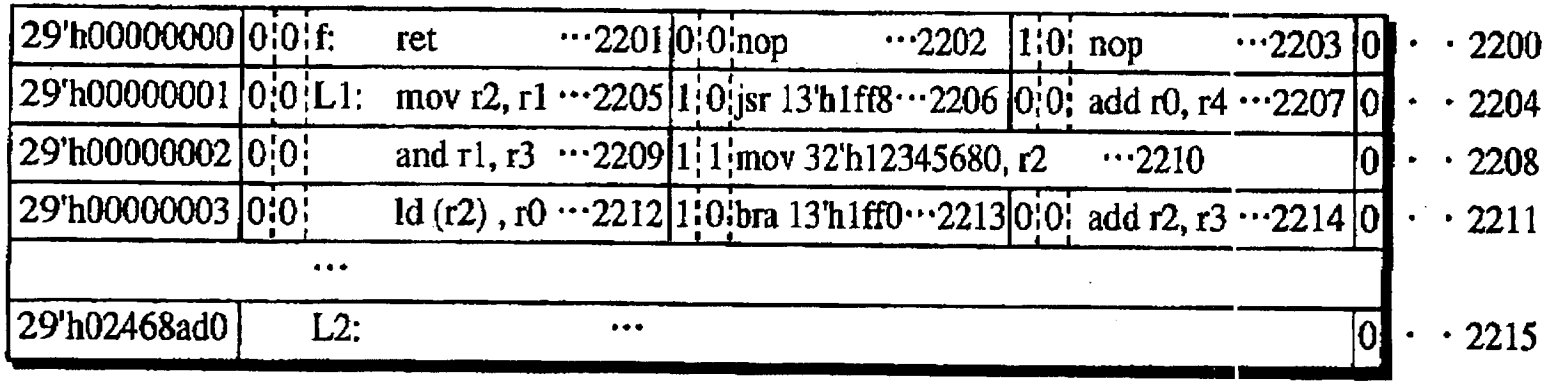

29'h00000000 0 0 f: ret ···2201 0 0 nop ···2202 1 0 nop ···2203 0 · · 2200
29'h00000001 0 0 L1: mov r2, r1 ···2205 1 0 jsr 13'h1ff8 ···2206 0 0 add r0, r4 ···2207 0 · · 2204
29'h00000002 0 0 and r1, r3 ···2209 1 1 mov 32'h12345680, r2 ···2210 0 · · 2208
29'h00000003 0 0 ld (r2) , r0 ···2212 1 0 bra 13'h1ff0 ···2213 0 0 add r2, r3 ···2214 0 · · 2211
···
29'h02468ad0 L2: ··· 0 · · 2215

FIG. 28A

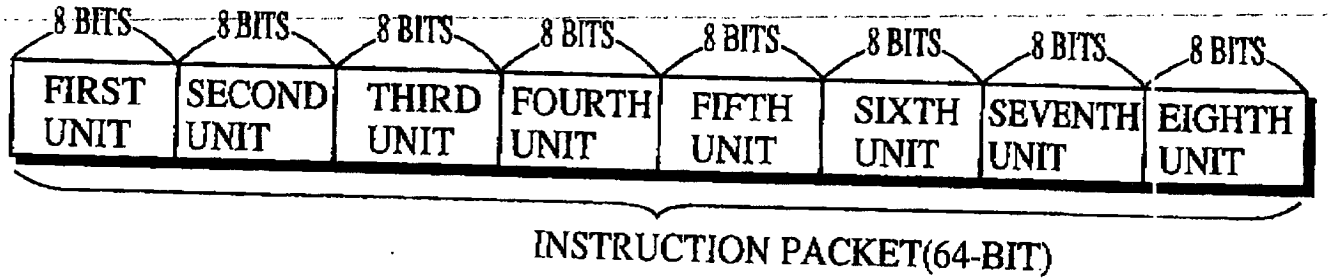


FIG. 28B

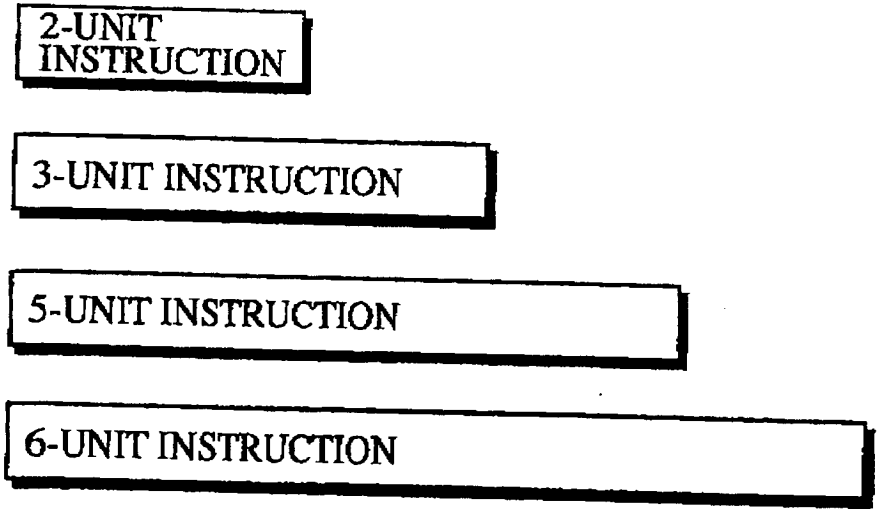


FIG. 28C

IN-PACKET ADDRESS	UNIT
3'b000	FIRST UNIT
3'b001	SECOND UNIT
3'b010	THIRD UNIT
3'b011	FOURTH UNIT
3'b100	FIFTH UNIT
3'b101	SIXTH UNIT
3'b110	SEVENTH UNIT
3'b111	EIGHTH UNIT

FIG. 29A

LOWER 3 BITS OF ADDRESS VALUE / LOWER 3 BITS OF PC RELATIVE VALUE	3'b000	3'b010	3'b100
3'b000	3'b000	3'b010	3'b100
3'b010	3'b010	3'b100	3'b000 (CARRY IGNORED)
3'b100	3'b100	3'b000 (CARRY IGNORED)	3'b010 (CARRY IGNORED)

FIG. 29B

LOWER 3 BITS OF ADDRESS VALUE (TO BE SUBTRACTED) / LOWER 3 BITS OF ADDRESS VALUE (BEFORE BE SUBTRACTION)	3'b000	3'b010	3'b100
3'b000	3'b000	3'b100 (CARRY IGNORED)	3'b010 (CARRY IGNORED)
3'b010	3'b010	3'b000	3'b100 (CARRY IGNORED)
3'b100	3'b100	3'b010	3'b000

FIG. 30

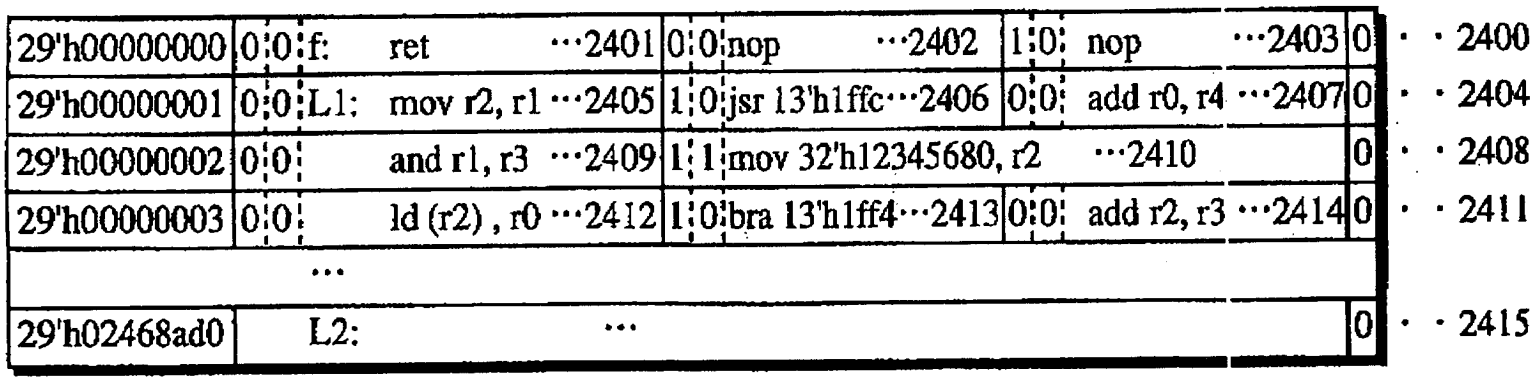

29'h00000000 0 0 f: ret …2401 0 0 nop …2402 1 0 nop …2403 0 · · 2400
29'h00000001 0 0 L1: mov r2, r1 …2405 1 0 jsr 13'h1ffc…2406 0 0 add r0, r4 …2407 0 · · 2404
29'h00000002 0 0 and r1, r3 …2409 1 1 mov 32'h12345680, r2 …2410 0 · · 2408
29'h00000003 0 0 ld (r2) , r0 …2412 1 0 bra 13'h1ff4…2413 0 0 add r2, r3 …2414 0 · · 2411
…
29'h02468ad0 L2: … 0 · · 2415

FIG. 31A

LOWER 3 BITS OF ADDRESS VALUE / LOWER 3 BITS OF PC RELATIVE VALUE	3'b000	3'b010	3'b100
3'b000	3'b000	3'b000	3'b000
3'b010	3'b010	3'b010	3'b010
3'b100	3'b100	3'b100	3'b100

FIG. 31B

LOWER 3 BITS OF ADDRESS VALUE / LOWER 3 BITS OF PC RELATIVE VALUE	3'b000	0b010	0b100
3'b000	3'b000	3'b000	3'b000
3'b010	3'b010	3'b010	3'b010
3'b100	3'b100	3'b100	3'b100

FIG. 32

29'h00000000	0	0	f:	ret	···2601	0	0	nop	···2602	1	0	nop	···2603	0	· · 2600
29'h00000001	0	0	L1:	mov r2, r1	···2605	1	0	jsr 13'h1ff8	···2606	0	0	add r0, r4	···2607	0	· · 2604
29'h00000002	0	0		and r1, r3	···2609	1	1	mov 32'h12345680, r2				···2610		0	· · 2608
29'h00000003	0	0		ld (r2) , r0	···2612	1	0	bra 13'h1ff0	···2613	0	0	add r2, r3	···2614	0	· · · 2611
			···												
29'h02468ad0			L2:		···									0	· · 2615

FIG. 33

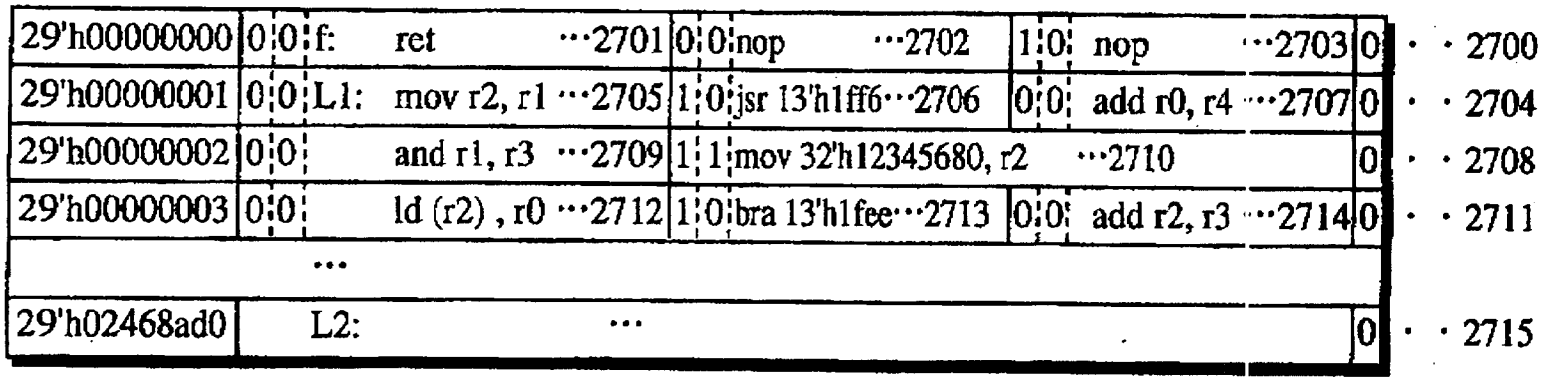
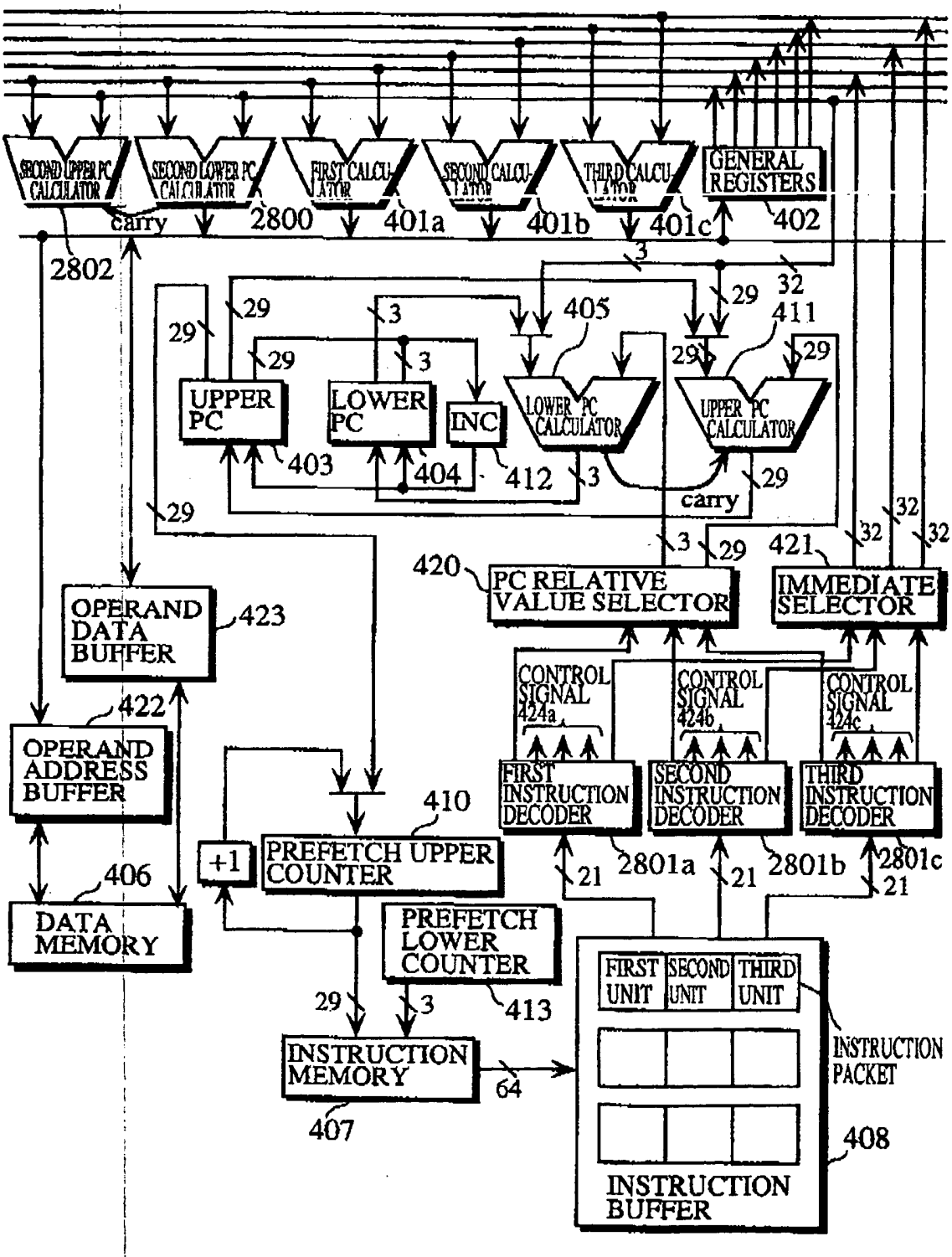

29'h00000000 0 0 f: ret …2701 0 0 nop …2702 1 0 nop …2703 0 · · 2700
29'h00000001 0 0 L1: mov r2, r1 …2705 1 0 jsr 13'h1ff6…2706 0 0 add r0, r4 …2707 0 · · 2704
29'h00000002 0 0 and r1, r3 …2709 1 1 mov 32'h12345680, r2 …2710 0 · · 2708
29'h00000003 0 0 ld (r2) , r0 …2712 1 0 bra 13'h1fee…2713 0 0 add r2, r3 …2714 0 · · 2711
…
29'h02468ad0 L2: … 0 · · 2715

FIG. 34


SECOND UPPER PC CALCULATOR
SECOND LOWER PC CALCULATOR
FIRST CALCULATOR
SECOND CALCULATOR
THIRD CALCULATOR
GENERAL REGISTERS
carry
2800
401a
401b
401c
402
2802
3
29
32
405
411
UPPER PC
LOWER PC
INC
LOWER PC CALCULATOR
UPPER PC CALCULATOR
403
404
412
carry
420
PC RELATIVE VALUE SELECTOR
421
IMMEDIATE SELECTOR
OPERAND DATA BUFFER
423
422
OPERAND ADDRESS BUFFER
CONTROL SIGNAL 424a
CONTROL SIGNAL 424b
CONTROL SIGNAL 424c
FIRST INSTRUCTION DECODER
SECOND INSTRUCTION DECODER
THIRD INSTRUCTION DECODER
410
+1
PREFETCH UPPER COUNTER
406
DATA MEMORY
PREFETCH LOWER COUNTER
413
21
2801a
2801b
2801c
FIRST UNIT
SECOND UNIT
THIRD UNIT
INSTRUCTION MEMORY
64
407
INSTRUCTION PACKET
408
INSTRUCTION BUFFER

	MNEMONIC	OPERATION
FIG. 35A	addpc disp , Rn	Rn + disp − > Rn
FIG. 35B	subpc disp , Rn	Rn − disp − > Rn

FIG. 36

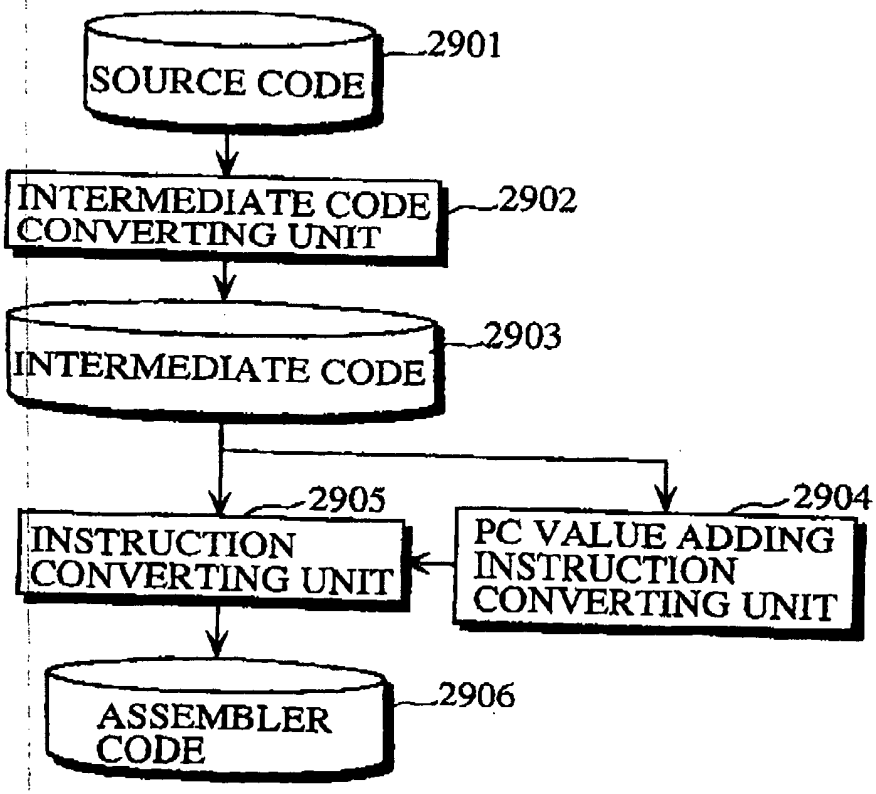


FIG. 37

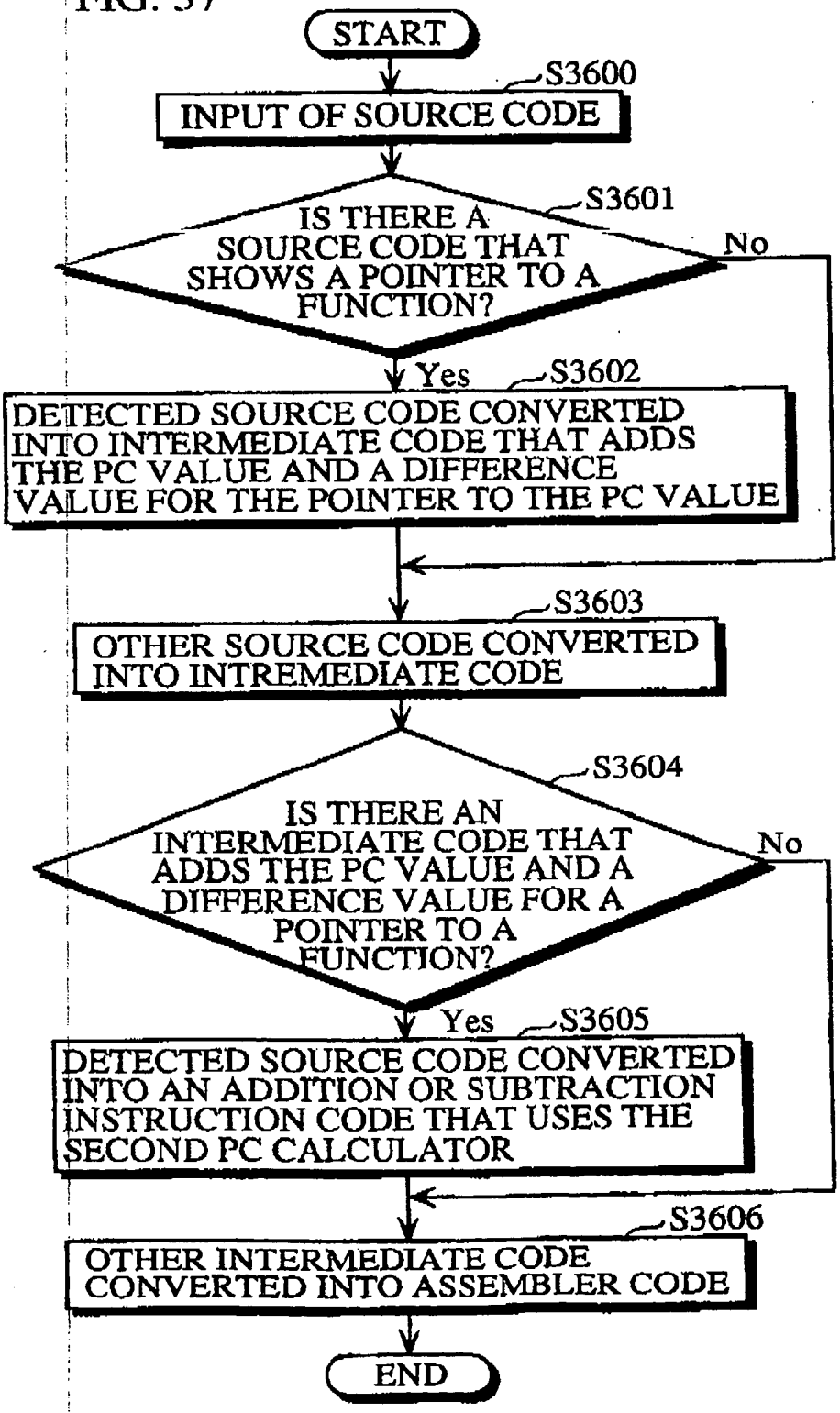

START
S3600
INPUT OF SOURCE CODE
S3601
IS THERE A SOURCE CODE THAT SHOWS A POINTER TO A FUNCTION?
No
Yes
S3602
DETECTED SOURCE CODE CONVERTED INTO INTERMEDIATE CODE THAT ADDS THE PC VALUE AND A DIFFERENCE VALUE FOR THE POINTER TO THE PC VALUE
S3603
OTHER SOURCE CODE CONVERTED INTO INTREMEDIATE CODE
S3604
IS THERE AN INTERMEDIATE CODE THAT ADDS THE PC VALUE AND A DIFFERENCE VALUE FOR A POINTER TO A FUNCTION?
No
Yes
S3605
DETECTED SOURCE CODE CONVERTED INTO AN ADDITION OR SUBTRACTION INSTRUCTION CODE THAT USES THE SECOND PC CALCULATOR
S3606
OTHER INTERMEDIATE CODE CONVERTED INTO ASSEMBLER CODE
END

FIG. 38

```
extern int g1 () ;
extern int g2 () ;
extern int g3 () ;
extern int g4 () ;

f ( int i )
{
        int (*fp) () ;

        switch(i) {
            case 1 : fp = g1 ;
                     break ;
            case 2 : fp = g2 ;
                     break ;
            case 3 : fp = g3 ;
                     break ;
            default : fp = g4 ;
        }

        (*fp) () ;
}
```

FIG. 39

```
f :     tmp = PC                    3201
        i  ! = 1                    3202
        br L1                       3203
        fp = (g1 − f) + tmp         3204
        jmp L                       3205
L1 :    i  ! = 2                    3206
        br L2                       3207
        fp = (g2 − f) + tmp         3208
        jmp L                       3209
L2 :    i  ! = 3                    3210
        br L3                       3211
        fp = (g3 − f) + tmp         3212
        jmp L                       3213
L3 :    fp = (g4 − f) + tmp         3214
L :     * (fp) (i)                  3215
```

FIG. 40

```
f :     mov     PC, r1          3201
        compne  1, r0           3202
        br      L1              3203
        addpc   g1−f, r1        3204
        jmp     L               3205
L1 :    cmpne   2, r0           3206
        br      L2              3207
        addpc   g2−f, r1        3208
        jmp     L               3209
L2 :    cmpne   3, r0           3210
        br      L3              3211
        addpc   g3−f, r1        3212
        jmp     L               3213
L3 :    addpc   g4−f, r1        3214
L :     jsr     (r1)            3215
        ret                     3216
```

FIG. 41

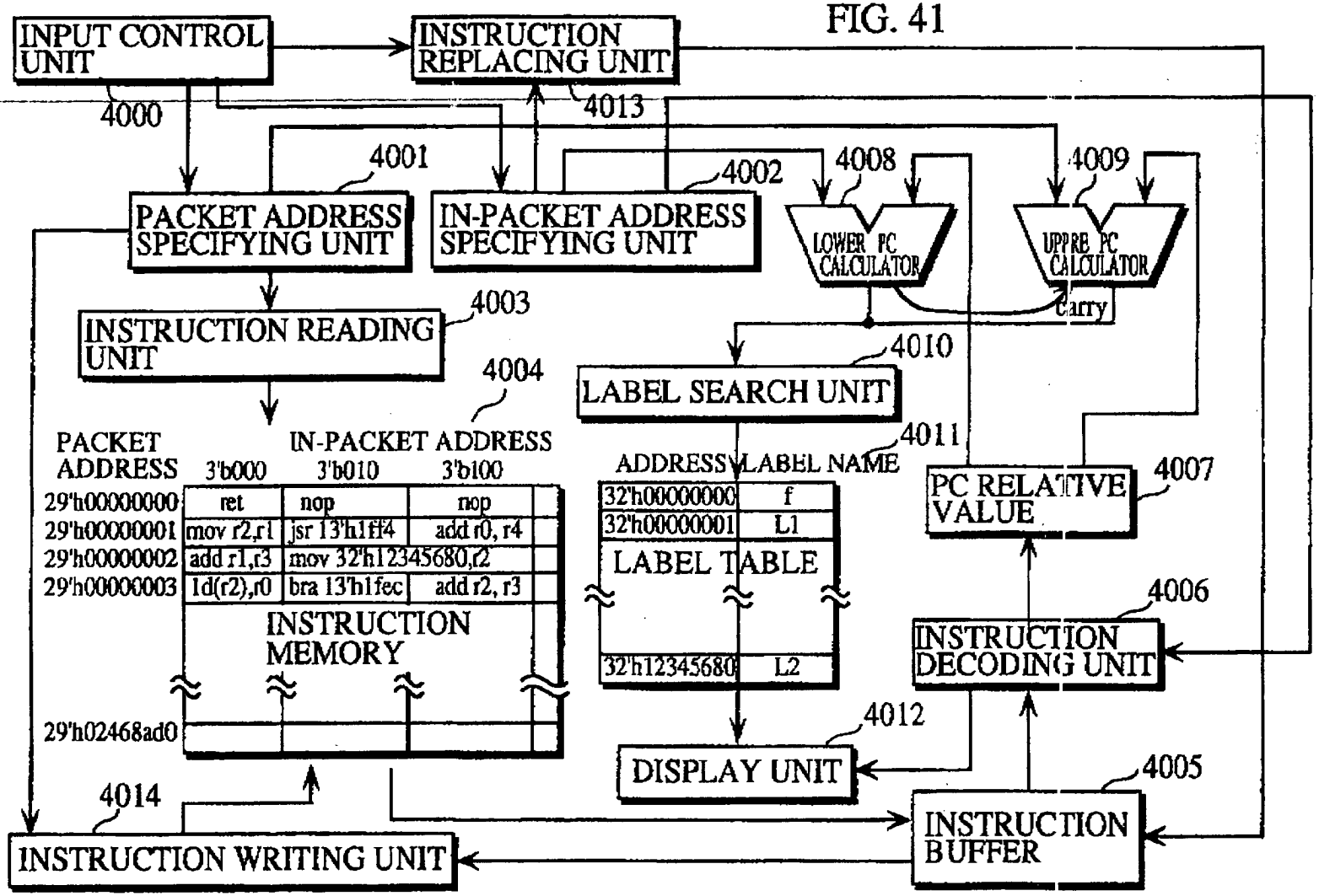

INPUT CONTROL UNIT
INSTRUCTION REPLACING UNIT
4000
4013
4001
4002
4008
4009
PACKET ADDRESS SPECIFYING UNIT
IN-PACKET ADDRESS SPECIFYING UNIT
LOWER PC CALCULATOR
UPPER PC CALCULATOR
carry
4003
INSTRUCTION READING UNIT
4004
4010
LABEL SEARCH UNIT
PACKET ADDRESS
IN-PACKET ADDRESS
3'b000
3'b010
3'b100
4011
ADDRESS
LABEL NAME
PC RELATIVE VALUE
4007
29'h00000000
ret
nop
nop
29'h00000001
mov r2,r1
jsr 13'h1ff4
add r0, r4
29'h00000002
add r1,r3
mov 32'h12345680,r2
29'h00000003
ld(r2),r0
bra 13'h1fec
add r2, r3
INSTRUCTION MEMORY
32'h00000000
f
32'h00000001
L1
LABEL TABLE
32'h12345680
L2
4006
INSTRUCTION DECODING UNIT
29'h02468ad0
4012
DISPLAY UNIT
4005
4014
INSTRUCTION BUFFER
INSTRUCTION WRITING UNIT

FIG. 42

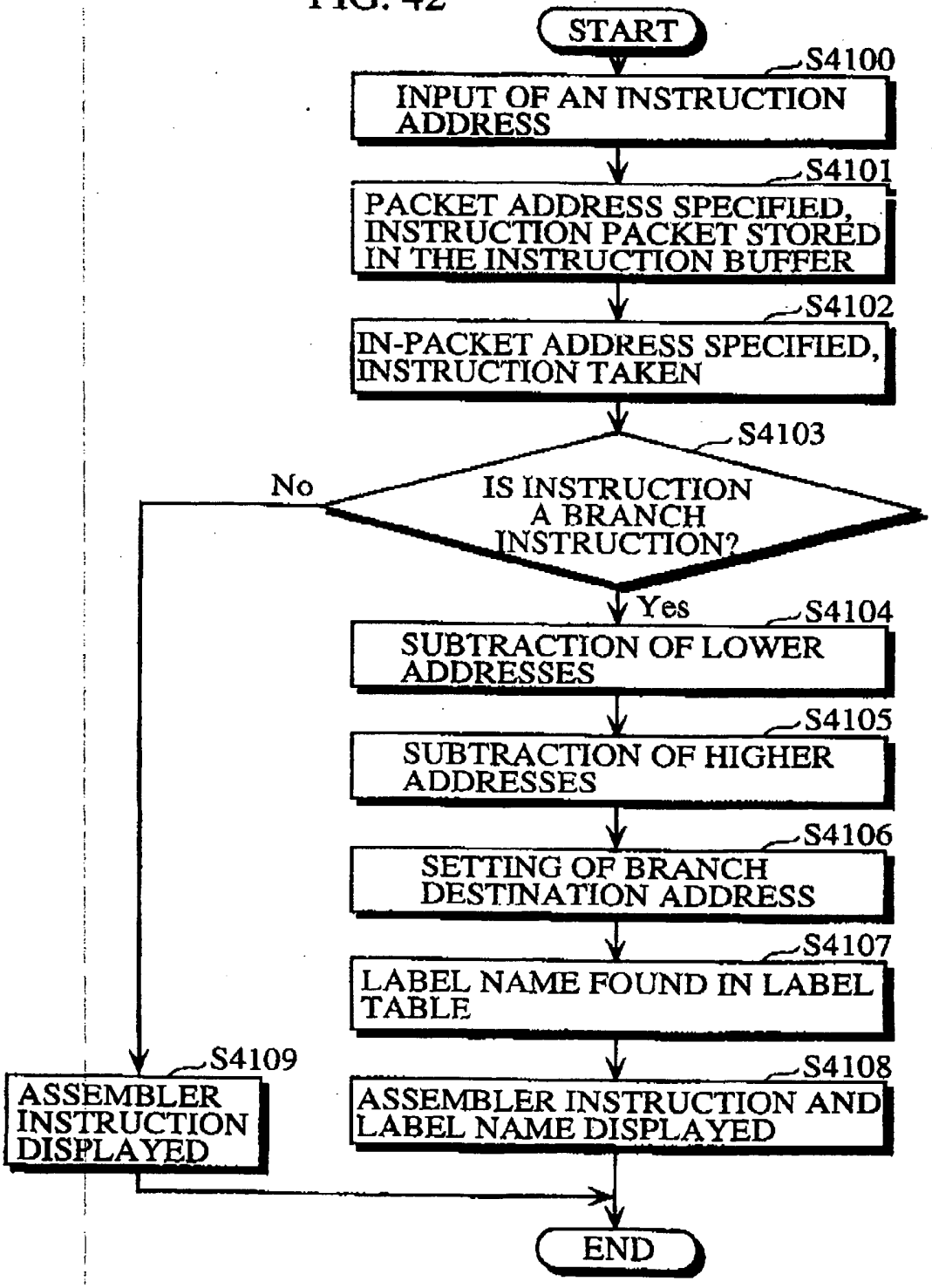
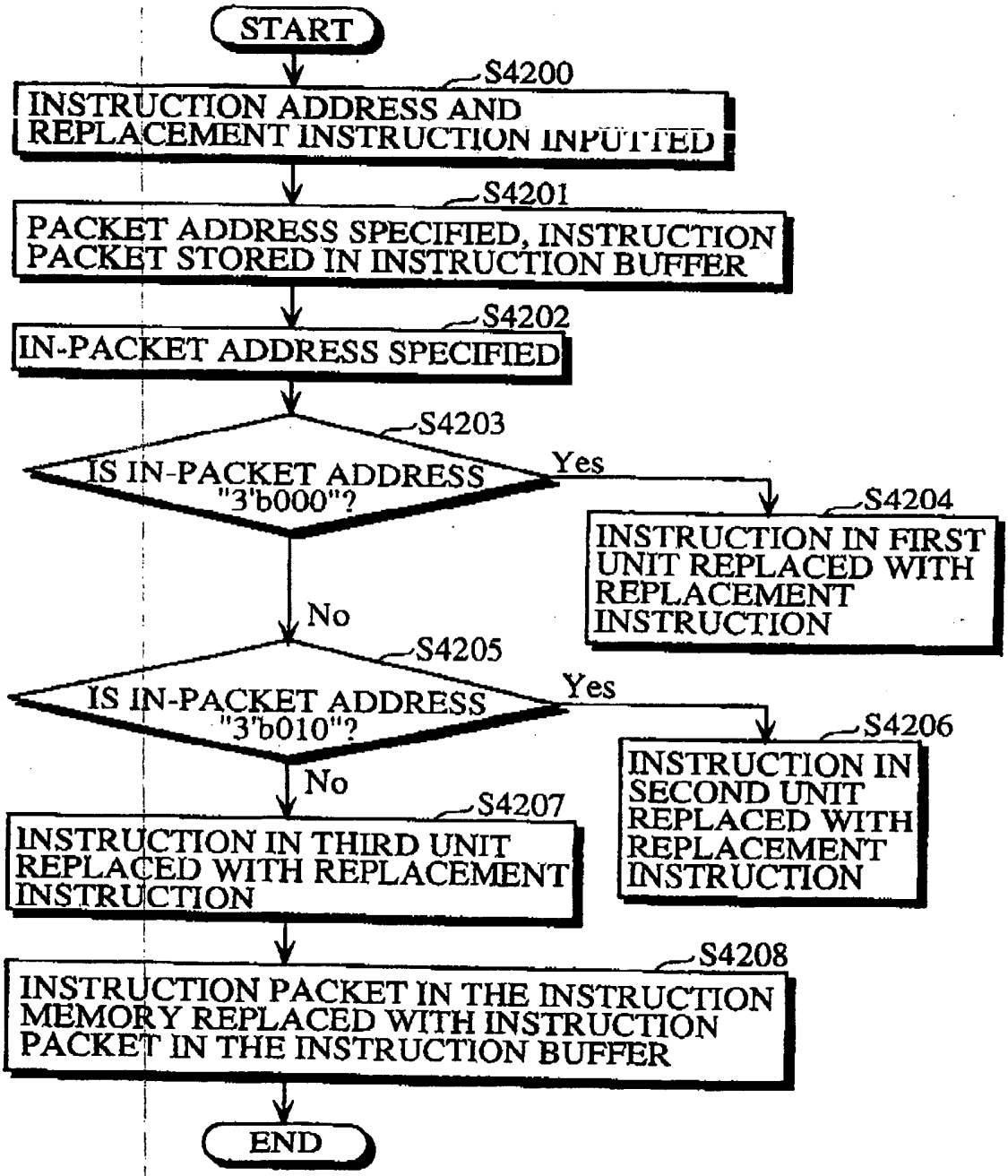

START
S4100
INPUT OF AN INSTRUCTION ADDRESS
S4101
PACKET ADDRESS SPECIFIED, INSTRUCTION PACKET STORED IN THE INSTRUCTION BUFFER
S4102
IN-PACKET ADDRESS SPECIFIED, INSTRUCTION TAKEN
S4103
IS INSTRUCTION A BRANCH INSTRUCTION?
No
Yes
S4104
SUBTRACTION OF LOWER ADDRESSES
S4105
SUBTRACTION OF HIGHER ADDRESSES
S4106
SETTING OF BRANCH DESTINATION ADDRESS
S4107
LABEL NAME FOUND IN LABEL TABLE
S4108
ASSEMBLER INSTRUCTION AND LABEL NAME DISPLAYED
S4109
ASSEMBLER INSTRUCTION DISPLAYED
END

FIG. 43


START
S4200
INSTRUCTION ADDRESS AND REPLACEMENT INSTRUCTION INPUTTED
S4201
PACKET ADDRESS SPECIFIED, INSTRUCTION PACKET STORED IN INSTRUCTION BUFFER
S4202
IN-PACKET ADDRESS SPECIFIED
S4203
IS IN-PACKET ADDRESS "3'b000"?
Yes
S4204
INSTRUCTION IN FIRST UNIT REPLACED WITH REPLACEMENT INSTRUCTION
No
S4205
IS IN-PACKET ADDRESS "3'b010"?
Yes
S4206
INSTRUCTION IN SECOND UNIT REPLACED WITH REPLACEMENT INSTRUCTION
No
S4207
INSTRUCTION IN THIRD UNIT REPLACED WITH REPLACEMENT INSTRUCTION
S4208
INSTRUCTION PACKET IN THE INSTRUCTION MEMORY REPLACED WITH INSTRUCTION PACKET IN THE INSTRUCTION BUFFER
END